A State Space Approach to Robotics

Daniel Slater

Elicon, Inc.
Brea, California

Abstract

Two robotic systems for the filming of motion picture special effects and the measurement of spacecraft antenna performance are described. Both systems are programmed in the FORTH language and use a common approach of state space methods to produce a highly modularized and reliable control program.

As the robot revolution continues, robots are taking on new and unusual applications. Two new applications discussed in this paper are in the area of motion picture effects filming and spacecraft antenna pattern measurements.

The Filming of Motion Picture Effects

The Elicon Special Effects Camera Robot

The costs of film production are rising rapidly and nowhere is this more apparent than in the area of graphics and miniature photography. With the introduction of films using high quality special effects, this task has become even more difficult. Elicon, Inc. has helped to solve this problem with the development of a robot camera system capable of producing high quality effects shots in a high volume production environment. [1]

The Elicon camera robot "flies" past models, building up images from scene elements (Figure 1). The scene elements consist of parts such as a starfield, spacecraft body, spacecraft running lights, and spacecraft window details. Each element would be filmed separately because of differences in size, lighting, setup, geometry or availability. Precise repeatability of the path of motion is required to ensure the correct juxtaposition of the scene elements.

The robot is physically configured as a boom arm supported camera on a long track. The boom arm configuration provides better camera maneuverability and less lighting interference than alternative configurations at the expense of more complex geometry transformation software. The basic configuration has 7 degrees of freedom including camera yaw, pitch, focus, and film advance, along with boom azimuth, elevation and track. [2, 3, 4] However, some systems have had as many as 24 axes.

The camera robot is controlled by a Digital Equipment Corporation (DEC) PDP 11/23 computer with 2 RX02 floppy diskette drives. The control program provides real time control of the robot servo motion in addition to handling operator communications, geometry transformations and move definition. The move is normally produced by using a special film language. This language allows the film maker to define the position and attitude of the image in the viewfinder relative to a moving target position. Both on-line and off-line programming is supported. The program was originally written in the polyFORTH language of FORTH Inc. using integer arithmetic for speed.

Clients using the Elicon system have done special effects for dozens of movies and commercials. As an example *Battle Beyond the Stars* and *Jaws 3D* both used the system. Titles for the movie, *The World According to GARP*, and *ABC Sports* have been done as well.

Spacecraft Antenna Pattern Measurements

The TRW Near Field Test Facility

Another robot is used at TRW for measuring radiation patterns of antennas mounted on the next generation of satellites. Some of these antennas are classified and others, like the new NASA 30/20 POC
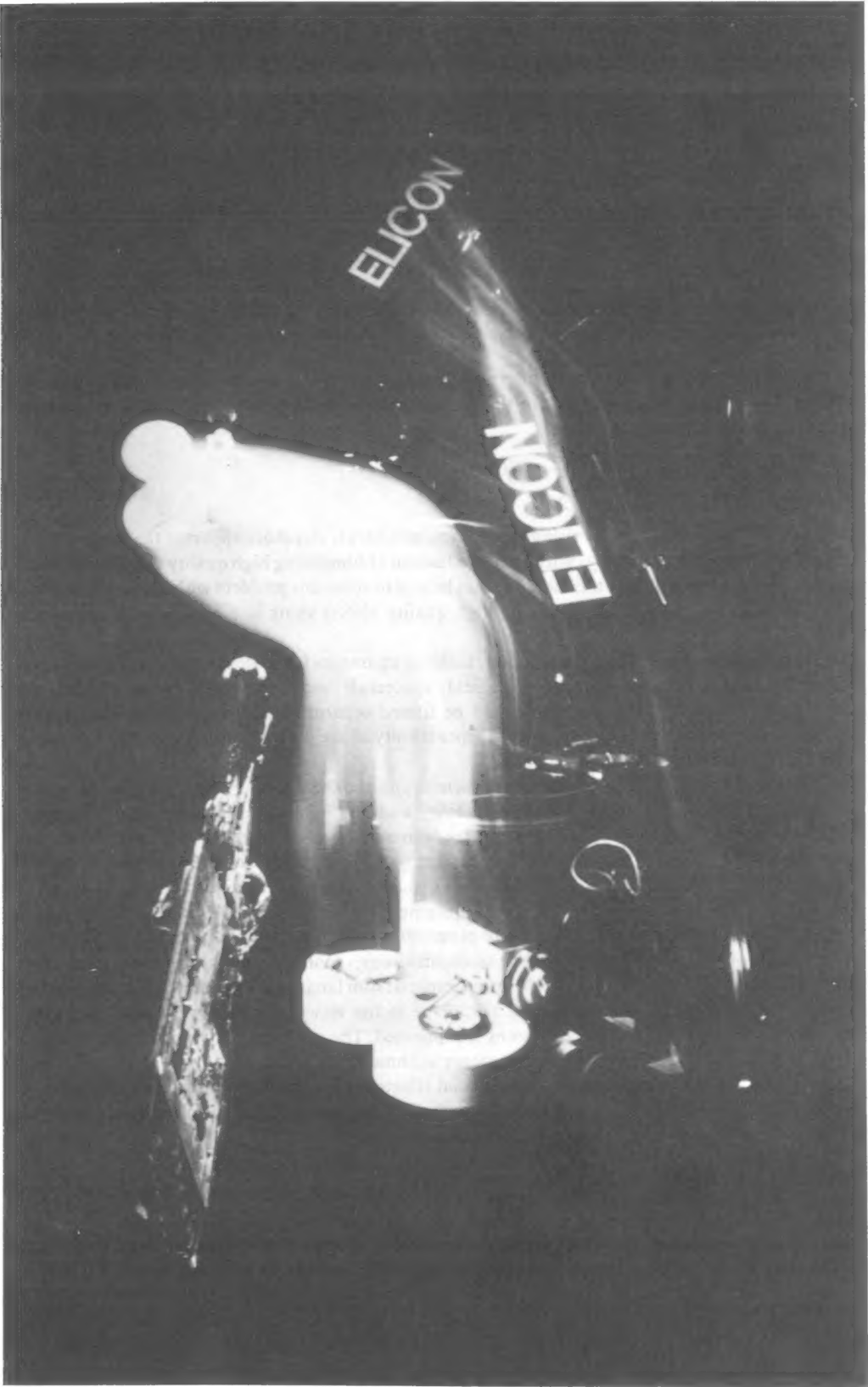
Figure 1. The Elicon Camera Robot moving past a spacecraft miniature.

antenna, will be used on commercial communications satellites. Unlike conventional antenna test ranges this facility operates in the near field of the antenna using mathematical methods to derive its far field performance. [5] This eliminates the logistic difficulty of measuring an antenna whose focus is several kilometers away, where there are atmospheric and ground interference as well as security problems.

The Near Field Test Facility (NFTF) physically consists of a probe antenna mounted on a 20 x 20 ft. servo controlled XY gantry, probing the radio frequency (RF) field of the stationary test antenna. (Figure 2) A VAX-11/780 computer system is used to perform the near to far field transformation. Positioning of the gantry is extremely critical, requiring that peak position errors be less than 0.001 inch over the entire scan range, or 1 part in 240,000. This accuracy is attained by a combination of kinematic structures, seismic isolation, thermal stabilization, laser interferometric position sensing and dynamic control of the scan command spectrum.
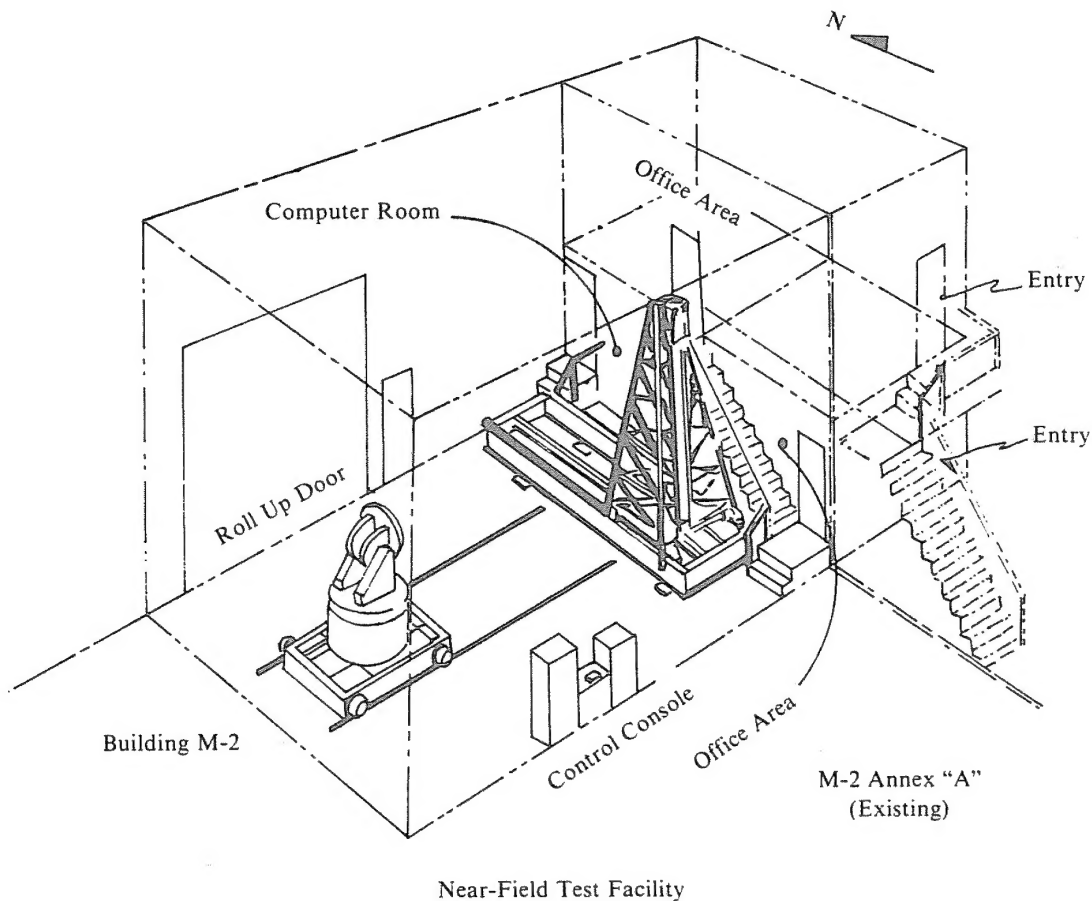

Near-Field Test Facility

Figure 2. Block diagram of the TRW Near Field Test Facility showing the overall relation between the test antenna and the precision scanner.

The control program for the NFTF operates on a PDP 11/23 computer. This program operates the RF scanner, the RF transmitter and receiver subsystem, and in general controls the data acquisition process. The control program is written in an enhanced, floating point version of polyFORTH. The program includes an extensive human interface to simplify operation of the NFTF. The control program includes a system integrity monitor serving as a watchdog for hardware and software faults and then, if requested, provides an analysis and recommended recovery of the fault to the operator.

Common Program Approach

The Robot Control Program Structure

Both systems share a common design philosophy, based on state space methods [6]. Central to the control program is a set of 4 vectors defined as follows:

1. State vector—The state vector is a minimum and non-redundant description of the system state. Elements include joint positions, limits, operating modes and other similar information.

2. Pseudo vector—The pseudo vector contains information redundant to the state vector for reasons of computational convenience. Example elements include the servo positions in a non-joint reference frame.

3. Measurement vector—The measurement vector contains sensor measurements from the real world. Examples include servo positions, errors and RF measurements.

4. Command vector—The command vector contains the commands for the robot.

These vectors contain most of the information that is usually carried in constants and variables. The TRW system state vector has approximately 250 elements, whereas the Elicon has over 500. Some of the information contained in a state vector can be seen in Figure 3.

The use of state and pseudo vectors allows for a "VisiCALC" like manipulation of the systems, and full simulation without hardware. Figure 4 shows the servo state, pseudo and measurement vectors.

The program uses a combination of heterarchical (parallel) and hierarchical (top down tree) structuring to produce a fast, simple and reliable program. The heterarchical structures are implemented as coroutines and are used at the highest level and many of the lower levels. An example of a heterarhical structure is the typical FORTH round-robin multitasker where no one task is more important than any other. Hierarchical structures are implemented as subroutines using the typical nesting of FORTH. The highest level of the program consists of coroutines which communicate only through the 4 data vectors. These coroutines include:

1. State vector editor—The state vector editor provides a human interface to the program, allowing the user to display and modify the system state. The editor presents the user's application using a tree structured menu. The internal operation of the menu system however, is heterarchical. The NFTF state editor is being extended to include voice input and output, allowing a single operator to control the NFTF. The voice I/O unit has both an originate and answer telephone interface allowing unattended operation with the ability to answer and page remote operators.

2. State to pseudo state conversion module—This module updates the pseudo state vector from the state vector. The pseudo state vector includes information redundant to the state vector such as the servo positions in alternative reference frames. This redundant information is stored for faster run time operation just as linkable binary object code is stored in some systems for faster code development. Most of this module consists of geometry conversions and servo dynamics modeling. This module is called whenever the state vector has been changed.

3. State vector initializer—This module initializes and saves the system state. The initialization is handled by copying a RAM and then initializing the pseudo state vector. The state vector may also be saved to one of the several areas on disk at any time for a checkpoint restart.

4. Servo handler—The servo handler causes the servos to move according to the positions and constraints established by the state vector. The servo handler reads the requested servo position from the state vector and applies it to a non-linear digital filter which establishes the desired servo dynamics. The filter output is then sent to the servo hardware to actually produce the motion. The servo handler also updates part of the measurement vector containing servo position and error measurements. Software development, training and debug are enhanced by the use of a servo hardware simulator allowing the program to operate on a spare computer. The simulator requires fewer than 10 lines of source code and is automatically configured into the program if the servo hardware is not present.

5. State transition module—This module propagates the state vector into the future using the command vector. The camera robot program uses precompiled time series files containing the robot trajectory. The NFTF program computes the trajectory on the fly using a world model of the antenna. In the NFTF program this trajectory may be previewed with a 3D graphics display.

6. Data recorder—This module handles data acquisition according to the configuration established in the state vector. The data recorder inputs may be dynamically reassigned to any of the data vector elements. The data inputs to the recorder are queued and then saved to local disk and/or transmitted

to the VAX computer for further processing. The data is queued to allow for latency in the disk or VAX data transfer. The data on the local disk may be simultaneously listed or parametrically plotted during the data acquisition process.

7. System Integrity Monitor—The SIM monitors all data vectors and determines the integrity of the hardware and software. If a fault is detected the SIM can provide expert advice on fault recovery. The faults are detected using both program traps and polling. The fault is then time stamped, logged to disk and displayed to the operator. The operator may then request an analysis of the fault with the computer supplying a fault definition, severity, side effects and recommended recovery procedure. If the system is operating unattended the system can attempt an automatic recovery. In either case the fault repair action is noted in the log file.

Multitasking

Both systems are multitasking, and are capable of 20 hz operation. The six tasks in the TRW system are:

Task 1: The servo system is controlled (4 channels) and the receiver is triggered by this task. Coordinate transformations and non linear motion filters are employed to handle system dynamics. The accelerometers and tilt meters are read and some statistics are computed.

Task 2: The receiver is read, some relationships are computed from the data, and the results are stored in data queues by this task.

Task 3: The data queues are sent to either, or both, disk and the VAX.

Task 4 and 5: Operator display functions are handled, and archived data can be played back while the system collects data in real-time.

Task 6: This task drives the hardcopy printer.

Conclusion

By using a highly modularized program operating directly on data vectors, program integrity, speed and compactness are enhanced. Both programs operate in a real time and reliable manner using less than 28K words of CPU memory and operate at 20 hz. The state space approach to robotics programming also allows for very complete hardware simulation. State space methods provide a powerful programming tool which has also been used for a flight simulator, a biomedical application, and a radar signal processor.

Acknowledgments

References

1. Prince, Suzan, "Elicon Camera Systems". *Computer Pictures Magazine*, July 1983.
2. Slater, Dan and Regla, Peter, "The Use of Precision DC Servo Systems in Motion Picture Special Effects Photography". *Drives and Controls International*, Vol. 1 No. 1, May 1981.
3. Slater, Dan, et al., "The Use of Computer Technology in Magican Slave Camera Systems". Vol. 49 *Proc. 1980 National Computer Conference*.
4. Slater, Dan and Pettus, Mike, "Image Synthesis with Servoed Cameras", Vol. 59 *Simulators and Simulation, Proc. Society of Photo-optical Instrumentation Engineers*, 1975.
5. IEEE, *IEEE Standard Test Procedures for Antennas, STD149-1979*, 1979; pp 59-62.
6. Dorf, Richard, *Modern Control Systems*, Addison Wesley, Reading, Mass., 1967; Chapter 9.

Mr. Slater attended California State University, in Fullerton, and studied physics and earth science. He is interested in computer graphics and artificial intelligence, and has published papers on servo control systems. Currently he is developing a synthetic aperture radar for TRW in Redondo Beach.

Figure 3. TRW State Vector

System Control Parameters

Scan Generator and Geometry Parameters
      yaw
      pitch
      roll
      x offset
      y offset
      z offset
      roll offset
      height of scan pattern
      width of scan pattern

Display Parameters

File Parameters
      block limit
      current record number
      min and max limits of data

Accelerometer Data Structures

Laser Interferometer Data Structures

Servo Data Structures

    *Most quantities in the TRW state vector are maintained as 32 bit floating point.

Figure 4. Servo State, Pseudo and Measurement Vectors

Servo State Vector

      #axes
      names
      current position
      current velocity
      scale factor
      index offset
      configuration data (on, off, enabled)
      forward limit
      reverse limit
      acceleration limit
      velocity limit
      calibration

Servo Pseudo Vector

      desired position in transformed coordinates
      filtered position   (filtering takes dynamics into account)
      filtered velocity
      filtered acceleration
      velocity per sample (instead of per unit time)
      acceleration per sample (instead of per unit time)

Servo Measurement Vector

      measured position
      measured error

   *The TRW NFTF has four servos